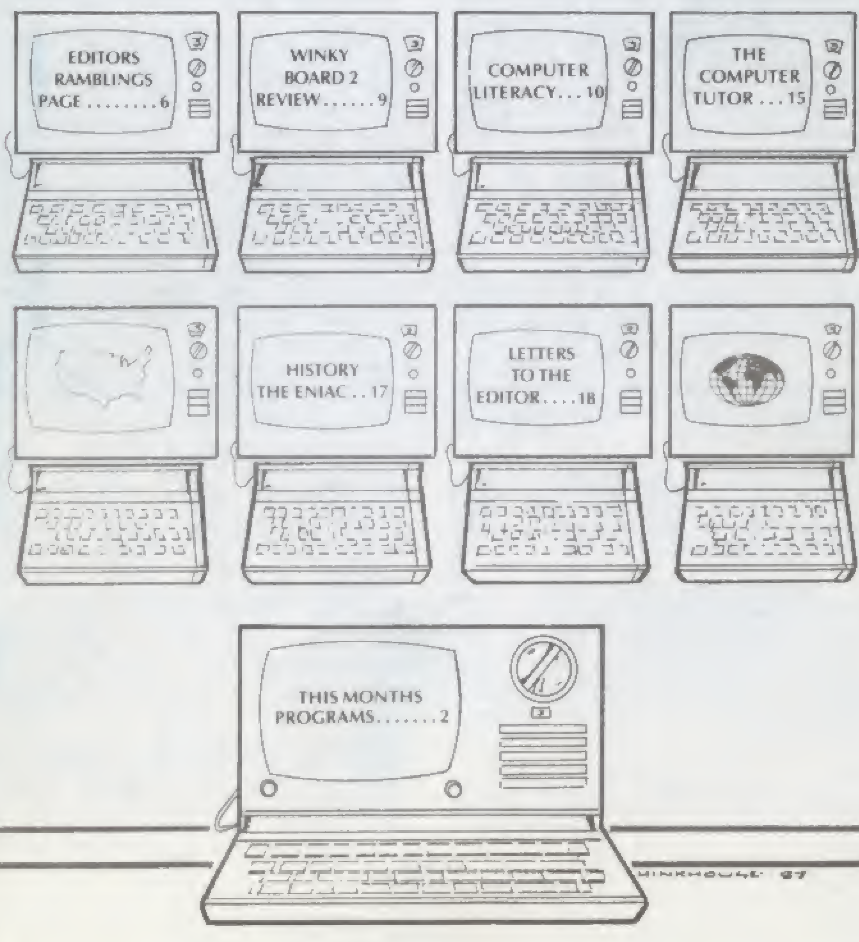
VOLUME .... 2 NUMBER 2

SYNCHRO·SETTE

THE SUBCRIPTION MAGAZINE FOR THE SINCLAIR ZX-81 / TS-1000 MICRO COMPUTERS

SYCHRO - SETTE IS PUBLISHED MONTHLY BY : THE S & S COMPANY
388 W. LAKE ST. ADDISON ILL. 60101 (312) 628-8955

IN THIS ISSUE FEBRUARY 1983



This Month's Programs

There are 8 programs on this month's cassette not counting the LOADER program.

	LOADER	- 2K
1 -	ADDITION	- 2K
2 -	SUBTRACTION	- 2K
3 -	WHITE-HOLE	- 2K
4 -	TAPESTRY	- 2K
5 -	PAYROLL	- 16K
6 -	SUPERSCROLL	- 16K
7 -	INVERSCROLL	- 16K
8 -	BULLETIN	- 16K

Each program is recorded only once on each side of the cassette. The first programs that can be LOADed will be of the small size followed by the larger sized programs. For example, this month's cassette has the first to fourth programs recorded as 2K programs. The 5th to 8th programs are 16K. The other side of the cassette is a duplicate of the first side.

For you new subscribers who aren't familiar with LOADing procedures for cassette programs, follow these directions:

A - Make sure that the volume setting of the recorder is set at about 80 % maximum.

B - If you have a Bass and/or Treble control on the recorder, make sure the Treble is at maximum and the Bass is at minimum.

C - To LOAD the first program, type in LOAD "" and press the ENTER key on the computer. Then press the PLAY button of the recorder. The lead time on the FEB/83 cassette is about 15 seconds until the first program begins.

The time needed to load the LOADER program is 26 seconds. When the program is loaded, a list of this month's programs will appear automatically.

Shut off the recorder when the LOADER program is loaded. Any of the listed programs can now be loaded into the computer by pressing the appropriate number on the keyboard and then pressing the PLAY key on the recorder. The loader program loads the program by searching for the name of the program you want and ignoring any of the other programs it may encounter along the way.

If you want to go directly to a program without waiting, we suggest you first find the tape location of the beginning of each of the programs with your recorder counter. This can be done as you go through the programs the first time, noting the tape location on the counter as each one is being loaded.

If you don't have a counter, approximate the tape position with the fast forward key just before where the program would start and then LOAD the program with the name of the program using the format LOAD "NAME OF PROGRAM".

Some of our subscribers have told us that they could not get the programs to load by name but they would load with the double quotes. Others have told us that the loader program wouldn't load certain programs. Most have told us that all the programs could be loaded either way. Every customer's cassette is made from the same master tape so the programs on everyone's cassettes are identical. We feel it is most probably a problem of volume adjustment or recorder design. We have noticed this situation on some of our recorders.

PROGRAMS (all programs this month are self running
- program's name has inverse last character /RT = run time/LT = load time)

There is an approximate 7 to 20 second pause between programs

LOADER 5-17

2/3 VOL

"ADDITION" LT = :26 18-28

All this little program does is add two numbers together. Great, huh! The unique feature of this program is that the numbers can be of any length of digits that can be accommodated on the screen. That's right, instead of 8 digit accuracy, you can have 800 or 5000 digit accuracy. The screen will fill up because of the program design, if the total of the digits entered for the two numbers exceeds about 400.

Observe the LISTing of this short program to see how the numbers are treated as string variables and the value of each character is manipulated.

1/2(+) VOL

"SUBTRACTION" LT = :24 29-40

This is the subtracting version of the one above. It uses a unique method of achieving subtraction through addition of reciprocals.

"ON (MARK)" VOL.

"WHITE-HOLE" LT - :28 41-53

Here's our big game for the month. It is the year 98,810,003,273 (July, 23rd). The universe is at the peak of its energy expansion and will soon start into its contraction phase (another billion years or so).

An unwanted result of this condition is the depletion of energy from the everyday (53 hour days) energy sources (magnetic gravity simulators).

The only way to replenish these sources is to bring back anti-matter in magnetic bottles and place them in the MGS storage receptacles thoughout the United Federation Galaxies.

Rich sources of anti-matter have been found in Quasars, which have been found to be White-Holes. Unlike Black-Holes that digest matter and anti-matter and make it disappear, White-Holes spew out matter and anti-matter.

Danger, of course, exists when matter and anti-matter come in contact and both are completely destroyed.

No known method has been devised to be able to determine from a distance whether an object is matter or anti-matter.

Your mission is to control a craft across the event horizon of a White-Hole and retrieve as much anti-matter as possible. Your craft is directed to the objects by pressing the <1> key and the <0> key for left and right. The length of the mission is 100 million miles. Your craft uses some of the anti-matter it collects for fuel.

The object is to collect as many tons of anti-matter as possible. When the craft touches an object, one of two things will happen:

1 - Nothing

2 - 100 tons of anti-matter will be absorbed and the rest will ve obliterated causing other matter or anti-matter in the vicinity to be destroyed.

For each million miles travelled, 1 ton of anti-matter will be used by the craft (very inneficient but due to the low ebb of the universe energy cycle).

A record is kept of the highest tonnage retrieved so that each pilot can have a record to try and beat.

(MARK+) VOL

"TAPESTRY" LT = :18 54-62

Ever wanted to design your own pattern for an Oriental rug you are chrocheting? Listen, if anyone out

there is good at that, let me know. I'll send you a thousand pounds of steel wool and you can knit me a Volkswagen.

Seriously though, this program generates a rectangular square pattern and then recycles and starts over again.

Insert line #10 RAND if you don't want to see the same pattern every time you fire the program up!

(MARK+) UCL

"PAYROLL" LT = 2:45 64-120

Our biggie for the month. The program is set up to accept 50 employees. If you want more or less, EDIT line #2 and change the <50> to whatever you want or what memory will allow. The higher that number is, the longer certain parts will pause in the program and vice-versa.

The program computes payroll under the following rules:

- 40 hour work week
- time and a half over 40 hours

Although this is by no means a program that is suitable for every payroll application, it is very powerful and has Visa-Calc type computing capabilities.

When the menu appears, first use prompt <7> to dimensionalize the variables. The program won't work at all until this is done.

Now, goto prompt <5> and set the tax levels. For the following inputs requested, enter whole numbers for the percentages. In other words, if six and a half is the percentage needed, enter <6.5>, not <.065>.

If there is no state income tax, enter <0>. When returning to the menu, you will notice a 5 second pause for 50 employees. This pause, and others you will encounter, can be shortened by dimensionalizing line #2 to the bare minimum needed.

Now go to prompt <1> from the menu and enter the required information. Observation of subroutine 8000 will help you determine the maximum amount of character or digit input for each data prompt. For those of us who want the easy way, they are:

EMPLOYEE NUMBER	6 CHAR
LAST NAME	12 CHAR
FIRST NAME	12 CHAR
HOURLY WAGE	5 CHAR
TOTAL HOURS	7 CHAR
OVERTIME	8 CHAR

Enter your own sample data for the displayed prompts.

For the prompt that asks for TOTAL HOURS you will notice that if the amount is in excess of 40 hours, two totals are split apart.

TOTAL NON-TAX DEDUCTIONS would be for payroll savings, stocks or bonds, loans or any other deductions that are not the taxes that were computed by prompt <5> from the MAIN MENU. You will, of course, have to total all these deductions yourself and enter the sum.

When the prompt to see the employee's pay occurs, enter "Y". You will notice that the items are all displayed except for the regular pay and overtime pay totals.

Well, we can't do everything for you! This is a good exercise for the user and I left enough space to the right of the titles for these data totals. All that remains is for you to write the routine (the Old Professor put me up to this).

Return to the MAIN MENU and execute prompt <3> and then enter the employee number of the sample you have entered.

You will notice a new EDITING MENU that allows you to edit any of the pertinent data except the tax variables. These can be changed by returning to the MAIN MENU later and going to prompt <5> and re-entering that data.

Try editing each of the data

items. Here's where the VISA-CALC type computing capabilities enter the picture. If you write down (if you have a printer, try writing a routine that will copy the screen from another number prompt here) the data for each situation, you will notice that each time data is edited, all the affected data is changed. Of course, the employer can use these editing features to experiment and see how a raise or proposed tax increase will affect his company's payroll picture.

This program has quite a bit of diversity in its use. The user can execute prompt <6> after all the employee information is entered and then SAVE the program on tape from prompt <4>. Now all the user has to do each week is recall that version of the program, edit the hours and the new payroll picture is presented.

Prompt <2> is basically the same as prompt <3> but without the editing features.

If new employees are added to the payroll, just use prompt <1> and the program will automatically allow the data to be added, as long as the total amount of employees does not exceed the limit allowed in line #2.

I didn't have time to write a routine to delete employees that leave the company but that would also be a good exercise for the user.

2/2 VOL

"SUPERSCROLL" LT = :35 122-135

Although this program could probably be written in 2K, the string manipulation routines use quite a bit of memory and would cause SCREEN FULL error messages.

A subscriber asked me if I could write a program that would scroll a message across the screen but in large letters so that he could use it to advertise his products. After thinking about it for a while, this is what I came up with using the BASIC language. The problem with this program will soon become obvious.

The only rule is that the words you enter cannot exceed 8 characters in length. There can be as many words as you want.

An excellent experiment for this program would be for a mother to type in a message or instructions for a small child who can read. It might keep him or her busy for hours.

3/5 VOL

"INVERSCROLL" LT = :39 138-152

Basically the same program as the one above, except white characters ar printed on a black background.

2/3 VOL 153-203

"BULLETIN" LT = 2:53 RT = 14:22

Our monthly Bulletin Board program. A unique feature of this program is if you want the scrolling message to stop, press the <P> key. To restart the message, press the <R> key.

You'll be glad we didn't use the techniques in programs 7 or 8 to scroll the message unless you have a few weeks of nothing to do on your hands.

TUTOR CONT.

two displayed totals. Add a line like: 10 REM or 10 PRINT

You will notice that these short programs occupy the same amount of space even though they have a different amount of characters in each line. Try back-spacing a REM before the PRINT in the line and ENTER GOTO 9990.

Why does the total not take up as much space as when they are entered as separate program lines? PRINT following a REM statement no longer has the characteristics as if it appeared alone. It now occupies only 1 memory location.

Type this routine in at the end of one of your existing programs but before executing it, put the computer in the FAST mode. The routine will count the memory locations at a rate of 1585 BYTEs per minute in the FAST mode. CONT. P. 14

Editor Ramblings

MISTAKES IN THE JANUARY/83 ISSUE

Phooey!
Page 10 - line #2030 should be:

```
2030 IF C$(LEN C$-2)="." THEN
    GOTO 2100
```

PAGE 14 - bottom left of page, delete lines 120 to 150. These lines were not omitted when the page layout was made.

PAGE 14 - line #180 should read:

```
180 LET J=I+1
```

NEW PRINTER FROM MINDWARE

A new printer will be offered from MINDWARE called the SIDEWINDER for $129.95. It is a dot matrix printer that will offer the same features and use the same paper as the MW-100 but will also print 80 columns. Besides printing in the established MW-100 format, it will also have the capability to print the lines sideways and according to my measurements this will give between 8 to 10 lines at a time. Portions of the printed data can then be pasted on a sheet of paper to give the desired layout.

A Mindware spokesperson told me that no delivery date is available yet but will keep in touch with us when information is available.

Sorry - upgrading the MW-100 is not available.

This same person told me that Mindware is gearing up to sell through retail outlets such as computer stores but will retain the mail-order service.

For further info on their products contact:

MINDWARE INC.
15 Tech Circle
Natick, MA, 01760
(617) 655-3388 *

SYNCHRO-SETTE TAPES NOW BULK PRODUCED

We no longer duplicate our own tapes for the bi-monthly subscriptions. Our subscriber size has made it impossible to for us to use our equipment for this purpose any longer. There are not enough hours available to do it. The tape you received this month was mass-produced by a company with high-speed equipment and quality materials.

It is an interesting process. When we were making them, we used tape recorders similiar to the ones you use and duplicated from one master recorder through pulse-regenerating and sound purifying equipment into ten slave or receiving recorders.

Average time for each subscription cassette was two and a half minutes.

Their process involves cutting a large roll of magnetic plastic film into thin strip reels. These reels are then fed into a machine which records, at high speed, the entire set of programs on each side of a measured section of the tape strip in both directions and then cuts it to the exact size needed and then inserts the tape, which is spun on

small spools, into the cases and seals the cases. Hundreds can be made in the time it makes us to make one.

The tapes have a blank segment about 15 seconds before the first program and after the last program. This of course makes for more efficiency to the user and and allows us to be charged for less materials. This is feasable when thousands of tapes are needed.

We will still, of course, use our equipment to produce the back-issue and stand-alone program cassettes.

COSMONICS UTILITY PROGRAMS

CFASTDATA and CFASTDUET are two new utility programs available for the TS/ZX computers.

- CFASTDATA provides all the capabilities of the Cosmonics READ/WRITE utility, but is 6 times faster. These programs occupy a small amount of memory space and can take data items such as those in a name and address program and store them on tape separate from the main program. This means you can write other programs that can use the same data and a tremendous amount of memory can be saved. Cost - 20.00 plus 1.50 shipping & handling.

- CFASTLOAD which has been available for a while, allows the ZX/TS to LOAD and SAVE programs and data (which has been saved with CFASTDATA) six times faster than normal. Cost - 10.00 plus 1.50 shipping & handling.

- CFASTDUET is the combination of the above two utilities on one cassette. Cost - 27.50 plus 1.50 shipping & handling.

The above utilities require a minimum of 16K RAM and Cosmonics recommends a recorder with a tape counter and quality cassette tapes. Short tapes will be OK in almost all situations for the recorded material because data and programs are stored on about 1/6 the of the space.

For further info, contact:

Cosmonics
PO BOX 10358
San Jose, CA, 95157 *

PRINTER INTERFACE

A smart interface that can connect standard serial or parallel port printers to the TS/ZX machines is now available from CRC software. It uses no RAM sells for $109.95. CRC also offers home, business and game software from 9.95 to 12.95.

For more info contact:

CRC Software
2901 Auburn Rd.
Auburn Hts., MI, 48057
313-852-3711 *

TOLL-FREE TIMEX NUMBER

Have a question about the TS-1000 or 2000 (they still don't know when the 2000 will be available)? How about cost information or technical assistance. Then call:

(800) 248-4639 *

TIMEX COMPUTER CLUB

FREE membership in this organization gets you a monthly copy of RAMBLINGS which will include regular early notices and a forum for members to swap ideas and achievements.

You supposedly have to be a member of a users' group. Tell them you heard about them in Synchro-Sette Users' magazine and that should do it.

Write to:

Timex Computer Club
PO BOX 2655
Waterbury, CT, 06725
Attn: Margo M. Murphy *

TWO NEW NEWSLETTERS

Bill -WINKY BOARD- Russell along with the Central PA TS/ZX Computer Users' Group is publishing SYNAPSE, a monthly newsletter for $10.00 per year. Bill asks for free material that is typewritten and ready for copy. In return for this material, you are assured of having the glory of seeing your name in print.

Bill is a very technical person and I expect to see answers to a lot of questions people have in these areas.

For subscriptions or submitted material, write:

Bill Russell - Editor
RD 1, BOX 539
Centre Hall, PA, 16828 *

::::::::::::::::::

TAS BAM (Timex And Sinclair Bay Are Microcomoter) Users' Group, formally known as SAM BAM (Sinclair And Microace Bay Area) Users' Group is publishing a newsletter called KEYBOARDS. No subscription rate or frequency of publication has been established as of yet but they were kind enough to send us a copy of their first issue.

They seem to place emphasis on sources of software, hardware, books and periodicals with the accent on unbiased information. Seems like a good place to find out about items of interest about your computer.

For further info contact:

TAS BAM User Group
PO BOX 644
Safety Harbor, FL, 33572

Mel Rout - Editor *

USERS' GROUP

In the New England Area, contact:

Worcester County Users' Group
Charles L. Kline
115 Francis Ave
Shrewsbury, MA, 01545
(617) 842-4163 *

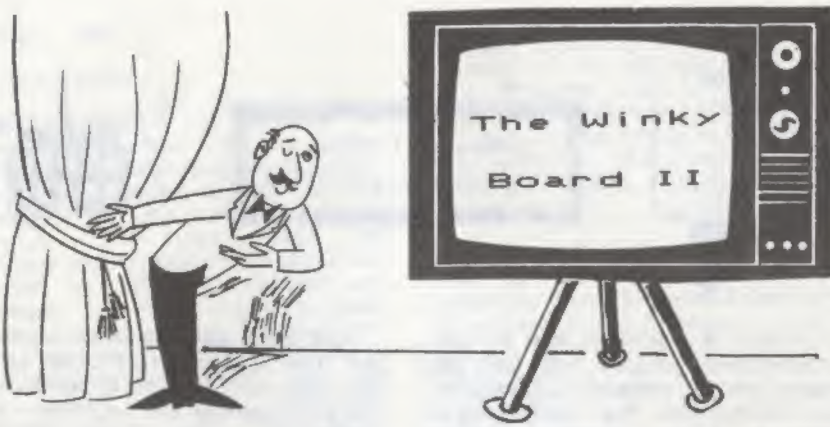
---

* TELL EM SYNCHRO-SETTE SENT YOU!

P.S.

YES, we will support the TS-2000 color computer when it becomes available - at first with articles and listed programs and within a few months with a new magazine using the same format as Synchro-Sette, 12 monthly issues, 6 bi-monthly cassettes. A tentative name will be "SYNCHRO-SETTE/2000".

NO, we do not have a TS-2000 or Spectrum yet although we had a chance to get the European Spectrum. We feel that the U.S. version will be significantly different and decided to wait until they become available here - when they do, you'll know it, and we'll tell you where to get them first - Ed.



WHAT IS A WINKY BOARD?

Ever wanted to duplicate a machine language program so that you would have a back-up copy? How about a tape that has a bunch of programs or routines on it that you wrote and would like to quickly make a copy for a friend? Let me tell you about an experience of mine!

We have, here at Synchro-Sette, tape duplicating equipment that allows us to duplicate multiple program tapes that we, ourselves, have prepared the master tapes for.

(Ed. note - it is unlawful, to make copies of copy.right- protected tapes that are to be used for anything other than back-up for the original purchaser.)

Preparing a master tape, particularly with both 2K and 16K programs, can be a nerve-racking experience. The sound quality has to be perfect. The 16K RAMpack can introduce background noise that can interfere with the reproduction process to a point where some of the duplicated programs may become distorted.

Combine this problem, along with trying to make more than one duplicate at a time, and the potential for defective reproduction multiplies.

The sound quality of a duplicate is usually not nearly as good as the master tape from which it was conceived.

Now we are given a tape with a machine language program on it that is of course a product of duplication, and confronted with the challenge of making a duplicate of the programs on it.

Such a challenge was presented with the ZX CHESS MASTER program. I had previously tried to make duplicates of it with our equipment but all attempts proved unsuccessful - that is until the WINKY BOARD 2 came along.

We originally didn't consider the WB-2 for this purpose because our equipment was set up to duplicate 10 multiple program tapes at a time and the WB-2 failed to consistently reproduce LOADable program tapes as well as the devices we were using.

We did notice at a later time, however, that on a one on one situation, the WB-2 not only gave consistently clean reproductions of BASIC programs but they were recorded at a much higher volume.

Before I go any farther, I must say that before discovering this effect, I couldn't get any reproduction of programs at all using the WB-2. A call to Mr.

CONT. P. 14

Computer Literacy In Elementary Schools

As many of you might know by now, I have been called both Hawk and Evangelist for computers in schools, particularly at the grade school level. It is my personal feeling that of all the computers available, the Sinclair machines offer the best teaching tool for youngsters. I base my beliefs primarily on two criteria - single keystroke command entry and cost efficiency.

Not only do my feelings reflect the Sinclair machines as being the best teaching tool at this time but also that computer literacy will make the difference in the earning power of the individual once he or she enters the job world.

Those who do not have exposure to computers at the elementary school level will be at a distict disadvantage compared to their counterparts who are presented a formatted computer literacy education in the early grade levels.

Consider how computers have already entered the picture in even the most basic levels. I am reminded of the story of an acquaintance of mine who ran a privately owned Fast-Food chain of restaurants.

The cash registers in each store, instead of having numbers on the keys, had pictures of hamburgers, cheeseburgers, polish sausages, etc. Soft drinks were entered by pressing keys that displayed LARGE COKE or MEDIUM TAB, etc. It was thought that this would eliminate almost any possibility of human error. The store employees were mostly high school students between the ages of 16 and 18.

One day a power blackout occurred and the cashiers, to fill te orders that they could, were forced to read the price menues and total the sales themselves. The quick-thinking manager went to a local store and purchased a bunch of pocket calculators and then instructed his employees how to total sales and figure out the correct tax.

After checking the hand written receipts that evening, he couldn't believe how many mistakes were made. Almost every sale, even ones for single items, had the tax either figured incorrectly or the total was not correct.

This told both of us something about the effectiveness of today's educational system. But, more important, one can visualize how the computer can take over simple tedious tasks and possibly, with either voice recognition devices or key entry from the customer, many of the human-held positions will be made obsolete.

This is not the type of computer operator one should strive to become.

Apparently I am not alone in this opinion. The following is a proposal prepared by the Urban Schools Collaborative from Northeastern University in Boston, Massachusetts for computer literacy in Elementary Schools.

Prepared by:

Thomas Clark
Terry Grobe
Gregory Coffin

January, 1983

I. Introduction:

Much is being written about the changing character of the labor force. High technology is currently the fastest growing segment of our economy. In the 1950's only 17 percent of the work force was involved in information occupations, whereas in 1980, 60 percent are involved in these jobs. In three years, according to an International Data Corporation survey, only one in five white collar workers will NOT use some type of electronic keyboard. Clearly, our country is in transition from an "industrial to an information society."

What does this transformation mean? Primarily, it means that service industries are replacing product industries. Factory occupations are becoming obsolete, and white-collar oportunities, robotics and high-technology manufacturing are the growth areas of the next decade.

The United States is in a bind - the unemployment rate of blue-collar workers remains and seems irreversible, yet severe shortages exist in high skilled, technical occupations.

The education system is responding to this challenge, albeit interventions are somewhat slow and uneven. In well-to-do suburbs, such as Scarsdale, NY, computer education started several years ago. Instruction now begins in the third grade and ends at the high-school level with two courses that "attract more than 200 students each semester."

The technology explosion threatens to create a gap between the rich and poor, between those that are "computer literate" and those that have no exposure to the medium. Bob Perlman, who directs the computer center at the Hubert Humphrey Occupational Resource Center says, "Out in the suburbs we have a situation where huge numbers of parents already work in the computer industry so their kids are children of the industry. That isn't true for our kids. They have no computers in the home, and they have no one in the family that works in the industry. They need to see computers here and understand them."

A survey conducted in October, 1980 by the National Center For Education Statistics found that only 50 percent of all secondary schools and 14 percent of all elementary schools had access to at least one microcomputer or computer terminal. Most students and teachers in the country are still computer illiterate.

In Boston some steps have been taken. There are 200 Apple computers in 30 schools - mostly high schools. Only one elementary school has more than one computer. Overall, only 15 percent of the city's youngsters have access to computer education.

This proposal seeks to establish a pilot computer literacy program for 5th graders at the Blackstone Community School. The program has a number of discrete but related goals:

1. To begin information/reasoning skill building early.

2. To expose students to a learning mechanism that has a high degree of attraction for them.

3. To promote racial harmony by involving students together in problem-solving computer work.

II. Objectives

This program suggests that grade 5 students can profitably study how to control and program simple computers in school so that the computer can become a medium of instruction and a learning tool for the rest of their academic lives. The computer can actively INTERACT with students, which gives it a significant advantage over books.

Because the computer can file, retrieve, sort, send, receive and print endlessly, students should learn the elements of these word-processing skills as soon as they have mastered the basics of reading and writing. Because a computer can calculate, compile and display graphics, students ought to learn how to manage the skill areas of data processing as soon as the basics of arithmetic have been mastered.

Both word-processing and data-processing skills can be learned by mastering a common computer language called BASIC (Beginner All purpose Symbolic Instruction Code). This is not the perfect language for all instructional purposes but it is ubiquitous and has been in existence long enough to have generated much useful software and support literature to promote its use. The Timex-1000 computer has an easily managed version of this language wired into its circuits and can be used to teach the basic techniques of computer operation and the elements of word processing.

Students in a computer workshop working with the Timex-1000 will master the following specific competencies.

Competencies:

1. Set up a Timex-1000 with:

a) antenna connector attached to VHF terminals on TV

b) 9-volt poower source plugged in

c) output connector to TV monitor connected

d) tape recorder connected to mic and ear outlets for memory storage

2. Describe the similarities with a typewriter keyboard, including:

a) order of letters of the alphabet

b) shift key

c) space bar/key

d) numbers

3. Describe the dissimilarities with a typewriter keyboard, including:

a) membrane type keys

b) number keys with Ø and O not being same

c) keyword option and function options

d) graphic options

e) presence of the ENTER key

f) presence of the delete option instead of backspace

4. Use the keyboard to:

a) print initial keywords

b) print words

c) erase letters and keywords using the delete key with shift

e) write words included in quotes

f) use the shift key to obtain all red letter options of keys

5. Execute in the immediate mode:

a) simple calculations with print command and enter

b) written outputs using print command, quotes and enter

c) numerical storage and retrieval using the LET command with storage location and equal sign and print command using storage location

d) a comma separated sequence

e) a semicolon separated sequence

6. Write computer programs that:

a) do simple calculation where the answer is associated with a printed statement of the problem

b) print out data that can be stored on magnetic tape, retrieved from magnetic tape, revised and re-entered on magnetic tape

c) display simple graphics

d) display data in simple tabulator form

7. Copy onto a computer previously recorded programs and then run them

8. Copy and/or enter from tape, previously written programs that include pupil-computer interaction

9. Recognize and remedy simple syntactic errors

10. Predict print-outs from examination of a program list

11. Debug a simple computer program

12. Define or describe the meaning or function of the following words and/or computer commands:

a) LIST	b) ENTER	c) DELETE
d) keyword	e) REM	f) LET
g) PRINT	h) CLEAR	i) INPUT
j) $	k) ,	l) ;
m) "	n) STOP	o) BREAK
p) IF	q) THEN	r) GOTO

III Activities

Fifth grade students will work one hour sessions twice a week for twelve weeks. The instruction will be provided by Thomas Clark, Professor of Education at Northeastern University. The instruction will also include a parallel workshop for teacher volunteers so that teachers on-site can continue to provide instruction when the time period provided by the grant expires.

The student workshop provides for a maximum of 20 students to work two on a machine for two one-hour periods each week. It also suggests that supervised practice time on the machines be provided at other scheduled times during the week. The curriculum (a teacher's guide to instruction) is included in an appendix to this proposal and was created specifically to accomplish the objectives previously outlined.

Ten Timex Computers (and TV screens), two tape recorders and 2 16K memory modules will be set up for use in a classroom where access and security have been established. A classroom teacher will assist in instruction, and a volunteer will supervise practice for students at times other than instruction times. Posters, diagrams and other instructional materials will be created or purchased for this workshop.

IV Budget

10 Timex-1000s	@ 80.00	- 800.00
10 Sanyo B/W TVs	@ 80.00	- 800.00
2 Tape recrds.	@ 35.00	- 70.00
2 16K RAMpacks	@ 45.00	- 90.00
Instructional materials		- 100.00
Teacher's fees		- 1000.00
Total		- 2860.00

V. Evaluation

All students will be given a post test on their abilities to perform the competencies outlined in this proposal. It is assumed that none will have any of these skills prior to instruction. In addition, students will be given a pre- and post- semantic differential test to see if their perception adequacy has changed, based upon awareness of mastery over the machine.

Submitted to Synchro-Sette by;

Gregory Coffin
Director
Urban Schools Collaborative

Very comprehensive! This outline can be used as an model by almost any institution of learning to create a teaching process to create computer literacy in our schools. I can think of no other method that can achieve this result with the cost-efficiency as outlined. If anything, the cost would probably be cheaper because of school discounts, bulk purchasing and price dropping of the various units. One major chain is selling the TS-1000s at less than $60.00 each with the factory rebate.

The only drawback I can see at this time is the lack of educational instructional software and text material geared to the TS-1000. If schools accept this outline or some form of it as being feasible, all you aspiring software authors out there might be realizing a very lucrative career.

Software subjects at first would center around computer-awareness topics. Within a short time, the classical subjects would enter the picture as mathematics, spelling, geography, history, etc. could be taught with computer aid.

One of the problems with present teaching methods is that no two teachers teach alike. Some teachers can present ideas to students in ways and with manners that allow students to grasp and comprehend ideas and concepts better than other teachers can. On other subjects, the other teachers may excel over the first group.

Using the computer as a tool with properly prepared text, the most successfully established teaching software programs would be administered to the students by the teachers in a uniform program that could, with time, only be improved. Any student would be more assured of receiving an education of the same caliber as a student in a class with the same subject but taught bv a different instructor - Ed.

TUTOR CONT.

The program PAYROLL is 5995 BYTEs in size and takes 3 minutes and 47 seconds for the count.

Oh, there's the bell - I think I'll try this pizza now.

HEY, THESE AREN'T ANCHOVIES - THEY'RE JELLY BEANS! WHERE IS THAT GUY?

WINKY CONT.

Russell cleared up this situation immediately. The computer ear jack of the WB-2 has to be plugged with either the supplied dummy plug or an un-connected computer cable plug when in the recorder-to-recorder duplicating mode. If this is not done, the LED lights as well as the duplicating process will not work at all.

After setting the master cassette volume level to the level prescribed in the documentation, I proceeded to make one duplicate. After this was done, I loaded the duplicate back into a computer that had been set up to allow the user to eavesdrop on the recorded sound pulses through a pair of headphones while the program was loading into the computer.

Voila! The frequency pulses sounded extremely clean (noise free and undistorted) and had excellent volume. The program loaded and executed perfectly.

This device would probably work well with 2 or 3 slave recorders hooked up in parallel but when 10 were used, the distortion effects could be seen.

Mr. Russell told me that within a few months, he will have available a SUPER WINKY that will have output and be designed for 10 slave recorders. Good news for you aspiring software vendors.

For further information contact:

G. Russell Electronics
RD 1 BOX 539
Centre Hall, PA, 16828

or call Bill Russell
814-364-1325 eves.

the Computer Tutor

Determining Program Size

Good morning class and happy Valentines Day. Thank you for all the cards and candy.

One thing I'm wondering is who left the heart shaped anchovy pizza on my desk? I might have known - the gentleman in the rear has a smile on his face.

Today's lesson involves finding out how large a program is that is entered into the Timex Computer. Does anyone know how this can be accomplished?

The man in the rear says that we can add up the sum total of all the characters in the program and this will correspond with the amount of bytes or memory spaces used by the program.

Although he is on the right track, this method will not give us the correct answer, the reason being that some commands require more or less bytes than the amount of characters in the spelling of that command. Also, the computer automatically injects its own code that does not appear in the listing.

Some of us have seen routines that PEEK two adjacent memory location addresses and multiply one of them by 256 and add that figure to the other. Many people are confused by this procedure and cannot comprehend how the counting of memory locations used by the program is done in that manner.

There should be a method of just counting the memory locations from where the program starts to where it ends.

Have no fear, there is a 9-line program technique that can be added to any program that has space for 9 more lines, and it will figure exactly how many memory locations or BYTES a program occupies in memory.

First of all, let me state that the first command or statement in the first line of a program has its character code residing in memory location 16513. The rest of the program takes up the memory locations immediately following each other.

All we have to do is to write a

routine that will count the memory locations, starting with 16513 until it reaches the end of the program, and subtract the amount of memory locations used by that routine.

In an earlier lesson we discussed the unique properties of a program line such as:

```
9990 REM COPY COPY
```

If you try to ENTER these three commands as listed, you will notice that when the first <COPY> command is attempted to be entered, the letter <Z> appears. We have to be sneaky here to be able to enter this format.

First enter <9990> and then <COPY>. Now back-space one position by using the back-arrow key (shift <5>). Now enter <COPY> again. Back-space one more position and enter <REM>. Now press the ENTER key and the line will be intact. Neat, Huh?

This line acts as a stop-gap to end the counting procedure as the last program memory location is encountered. Again, as explained in the Syntactic Sum tutorial article, the computer is instructed to ignore anything that appears in a line that starts with a <REM> statement. The two <COPY> commands that follow provide a unique situation that is impossible to achieve in a normal programming situation.

The Sinclair character code number for the command COPY, as listed in Appendix A of the TS-1000 manual, is 255. This means that if a program line has two COPYs right next to each other, the sum total of the two character code amounts is 510. Since it is impossible to have the computer accept these two commands right next to each other for any situation except after the REM statement and since the number is the largest possible number that can be achieved by adding any two character codes together, we are safe in assuming that this format will accomplish our task.

The rest of the program routine is simple. The entire routine is as follows: USE 'FAST'

```
9990 REM COPY COPY
9991 LET N = 16513
9992 LET A = 0
9993 IF PEEK N + PEEK (N+1) =
     510 THEN GOTO 9997
9994 LET A = A+1
9995 LET N = N+1
9996 GOTO 9993
9997 PRINT AT 10,0;"TOTAL BYTES
     IN PROGRAM = ";A-1
9998 PRINT ,,"TOTAL SIZE OF
     PROGRAM IS ";100*INT (((A-1)
     /1024)+.05)/100;"-K"
```

Line 9991 sets the variable <N> to be the first memory location where the program starts. Line 9992 sets the variable <A> to be 0. Variable <N> is going to increment itself by one for each memory location that is being counted. Variable <A> is also going to increment itself by one each time <N> does.

Variable <A> is actually counting the memory locations or bytes in the program where variable <N> is the address of these memory locations.

A continuous loop occurs between lines 9993 to 9996 which line 9993 directs a breakout to occur to line 9997 when the contents of the current memory location being observed, and the one adjacent to it, have the total sum of their character codes equal 510.

Line 9997 then takes the total bytes counted, which is the variable <A> at this point and subtracts 1 from that total. The reason it does this is because if you observe line 9990, you will see that the first thing that appears in that line is the REM statement, and this occupies 1 memory location. Since this REM begins the counting routine and is not part of the program itself, we don't want it to be part of the final total.

Execute this routine by itself by typing in GOTO 9990. You will notice that <0> will be the result for the

CONT. P. 5

HISTORY
The Eniac

An ingenious idea, called a computer, has helped us do mathematical figuring a lot faster. The first electronic computer, built in 1946, had 18,000 vacuum tubes and was so large it filled an entire room.

The ENIAC (Electronic Numerical Integrator And Computer) embodied nearly all the components and standards that became standard in later machines. It was the monster prototype of the modern computer. It weighed over 30 tons, had 30 separate units and occupied a space 30 by 50 feet. It had some 18,000 vacuum tubes, 70,000 resistors and 6,000 switches. A synchronous machine with a clock rate of 100,000 pulses per second, the ENIAC had 20 accumulators for addition and subtraction, a multiplier and a combination device that performed square roots and division.

Data was transferred to and from the user with punched cards for both input and output.

The machine could identify the sign of a number, compare quantities for equality, add, subtract, multiply, divide and extract square roots. It stored up to 20 ten digit decimal numbers but there was no central memory as such. Storage of data was localized within the functioning units of the computer.

The ENIAC was developed by John W. Mauchly and J. Presper Eckert and offered an improvement in speed of more than 500 to 1 over its predecessors such as the electro-mechanical MARK-1 and MARK-2, which operated with relays rather than vacuum tubes.

It has been said that the neccessities of war are the prime motivators of industry. The ENIAC was the result of such motivations. The scientific and logistic support of the U.S. Ordnance Department used the ENIAC and was staffed by a handful of officers and career civilians. Its only scientific facility was the Aberdeen Proving Ground.

One of the more important functions of the ENIAC here was the production of firing and bombing tables and related gun control data. A typical firing table called for several thousand firing trajectories.

A war might be over before they could be compiled by the methods used before the ENIAC came into being.

The ENIAC was finally assembled in the Autumn of 1945 at the Moore School in Pennsylvania, doing problems in ballistics and with atomic energy equations. It was then dismantled and moved to Aberdeen where it was de-bugged. It was only here where its true worth became apparent.

It had taken a skilled operator about 20 hours to compute a 60 second trajectory. The differential analyzer took about 15 minutes to do the same job. The ENIAC did it in only 30 seconds - half the time it took the projectile to reach its target.

The ENIAC was the premier machine up to 1952 when newer computers that were more powerful and economical were introduced. The introduction of the transistor was the death knell for the ENIAC and on October 2, 1955, its power was disconnected.

Sections of it are preserved in various museums. - Marilyn Buza

Letters To The Editor

Dear Ed,

How come that guy in the back of the room did not speak up when the Old Professor showed the class how to sort the second column of the double array, page 12 of your October (82) issue? Lines 1111 to 1113, as you printed them, sort, but with the spelling reversed. What good is that?

It seems to me that you have to go through the entire routine again with <C$> replacing <B$> to get a useable sort of the second column.

On the October/82 tape, REACTION program, there were several places where it contained extraneous, unedited lines and phrases. And, what would you say if a player reached the difficult 10 level, only to find that the program did not print out the congratulations as you intended?

Yours truly - W. Fricke, Depew, NY

Dear Warren,

I asked the Old Professor and he said he was suprised at you. The whole purpose of the reverse arrays was to show you that they were linked to the ones in the other column.

Think of it this way. If you wanted to sort names and phone numbers, you wouldn't want the names to be sorted in one column alphabetically and the phone numbers in another column right next to it, sorted numerically. You would have no way to determine which name was linked to which phone number.

The program generated one set of numbers and then another set which was a reverse or mirror image of the first set. To show that a sort routine can also keep the associated variables linked the set of sorted variables was the purpose the Old Professor was trying to point out.

The purpose of the mirror image variables, which were separate but linked, was so that the student could easily see that they were associated with the main variables.

As far as the extraneous, unedited lines and phrases are concerned, call it Writer's Prerogative or whatever, but when one is writing a particularly long program and debugs it to the point where it is working properly, it is not necessary to go through the listing and remove all the un-needed program line material unless it is going to be used for a tutorial. It is true that if a person had to copy the listing, he or she might wonder why certain line material existed, but that is not the case here since it was on tape.

This extraneous material was once part of a routine that was being tested to achieve a certain effect. It is not difficult to enter a whole bunch of line material where, when the desired goal is finally realized, the programmer is hesitant to remove any of the material, which in actuality is extraneous, because of the time involved or maybe just because of fear itself.

It is also possible that the programmer may be used to writing in a language from another computer that when written on that other computer, may require some of this extraneous material.

Call it sloppy if you want, but try writing a long program from scratch that uses a lot of mathematical routines and I'll bet in most cases, there will be some extraneous material.

Besides, if the desired goal is achieved, what difference does it make? With microcomputers, we are taught to be as conservative as possible because of memory limitations. Yet, we are envious of programmers of more intricate languages such as APL, COBOL, FORTRAN and PASCAL on such main-frame machines such as IBM, General Data, Honeywell, Sperry Univac, DEC and other computers where memory is virtually limitless.

Talk to just about any one of these programmers who owns a microcomputer and almost all of them will tell you that one of the biggest difficulties they have programming a microcomputer is the small amount of memory they have to work with. You see, they're used to wasting memory in the programs they write on the main-frames.

Their bosses want them to write programs that work. They don't want to pay them extra money to go over a working program and remove extraneous material. There's an old saying - Never fix anything that isn't broken!

In regard to reaching level 10 in the REACTION program - nobody has done it yet that I know of and I doubt very much if anyone ever will.

Besides, the Old Professor said the guy in the back row was asleep anyway - Ed.

Dear Ed,

Seems like a lot of people are having trouble with the printing paper hanging up in the Sinclair printer - including you.

I wonder if it is the original roll or the replacement paper that is causing the problem. I had the same thing happen to me until I made an interesting discovery.

The replacement paper (Gladstone) is a fraction of an inch wider than the paper that was in the printer. To compound the problem, the paper was not wound on the spool with the spool, square with the paper, making it still wider.

In my case, it was a matter of rubbing off the protrusion of the plastic spool on sandpaper or emery cloth and the problem was cured. My sister had the same problem with her printer and solved it the same way. Dennison (paper manufacturer) must have some weird ways of rolling their paper, the spool end being inset on one end and protruding on the other.

Try it - see if it helps. I haven't had to pull paper since and I'm on my third spool.

Sincerely, G. Cary - Coloma, CA

Dear Mr. Cary,

I did, and it doesn't - work, that is.

We have two Sinclair printers and both share the same problem. The paper easily shifts back and forth on the roll with little pressure, but upon observance under operation, this seems to have nothing to do with the hang-up problem I am experiencing.

The little rubber roller, that is responsible for advancing the paper, doesn't seem to be creating enough friction - Ed.

Dear Ed,

In the October/82 issue, page #11, the Computer Tutor, the Shell Sort program (#3), I have found an error. if 55 or more stings are sorted. two or more are left in the wrong order. On the second pass of the sort, all will be in the right order.

As an experiment, I changed the logic operation in line 1080 from <= to <. With this change, the sort worked perfectly every time for all

amounts of strings. I found the same problem in the FIELD SORT program in the November issue. I would appreciate any comments on why the problem occurs and why the fix works.

C. Stephenson - Salt Lake City, UT

Dear Chuck,

"I dunno! Works fine on other computers - been using it for years. Actually, the program I plagiarized it from, years ago, probably had it incorrectly listed and since I was only interested in sorting small amounts of variables, I never encountered any problems - who is this guy anyhow?" is what the Old Professor told me. I guess he doesn't like to be embarressed - Ed.

Dear Ed,

Just got my first issue of Synchro-Sette and was impressed. However, I'm wondering if there is an error in the Bar-Graph program on page 14 (12/82)? When I ran it as printed, the input tabulation remained on the screen behind the graph. I was able to fix this problem by adding line 115 CLS. This might help others who had this problem.

J. Cable - Lehigh Acres, FL

Dear John,

I won't tell you what the Old professor said about this one. At first he tried to cop-out by saying that the user could use the old display to correlate the displayed data but anyone can see the messy screen display is better off without the data. Besides, the data is displayed on the same page (14) anyway.

Keep up the good work - all of you - we'll put him in his place yet! - Ed.
